

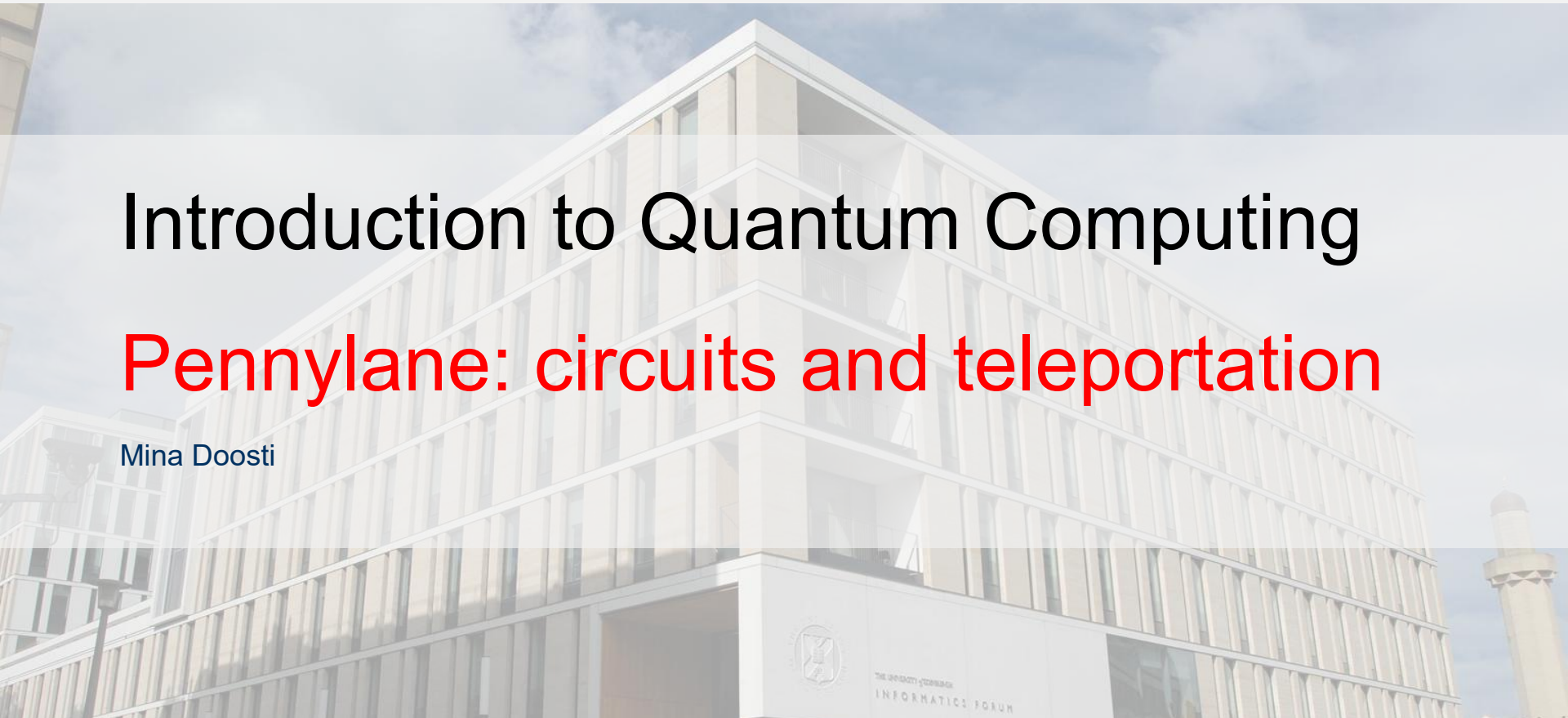


THE UNIVERSITY of EDINBURGH
informatics

Introduction to Quantum Computing

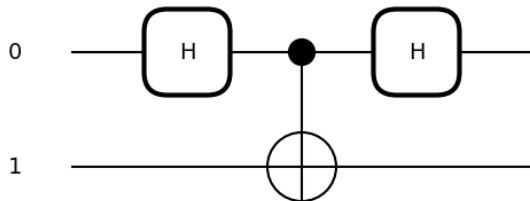
Pennylane: circuits and teleportation

Mina Doosti



Exercise 1: Circuits

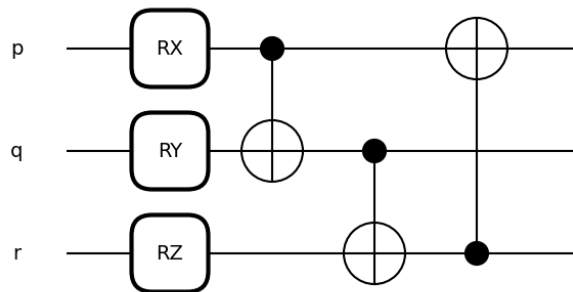
```
import pennylane as pl
from pennylane import numpy as np
import matplotlib.pyplot as plt
```



```
def circuit():
    # Call pl.Hadamard and pl.CNOT to build and return this circuit
    pl.
    return pl.

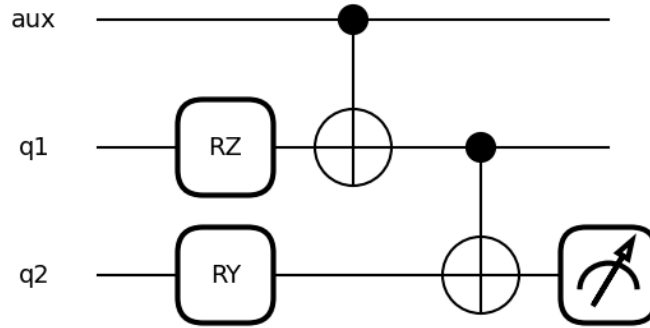
print(pl.draw(circuit)())
```

Exercise 2: Parameterised circuits



```
def circuit(alpha,beta,gamma):  
    # Call pl.RX, pl.RY, pl.RZ, and pl.CNOT to build and return this circuit  
    pl.  
    return pl.  
  
pl.drawer.use_style("black_white")  
pl.draw_mpl(circuit)(np.pi/4, np.pi/8, 0)
```

Exercise 3: Executing circuits



```
device = pl.device('default.qubit', wires=['aux', 'q1', 'q2'], shots=[1,10,100,1000])
def circuit(alpha, beta):
    # Build and return this circuit, using pl.expval and pl.PauliZ to measure qubit 2 in the Z basis
    pl.
    return pl.

qnode = pl.QNode(circuit,device)
qnode(np.pi/4, np.pi/4)
```

Exercise 4: Circuit depth

```
device = pl.device('default.qubit', wires=3)

@pl.qnode(device)
def circuit(x, y):
    # Build any circuit and inspect its depth
    return pl.

pl.draw_mpl(circuit) (0,0)
print(pl.specs(circuit) (0,0) ['resources'])

@pl.compile
@pl.qnode(device)
def circuit(x, y):
    # Build any circuit and inspect its depth, now using @pl.compile decorator
    return pl.

pl.draw_mpl(circuit) (0,0)
print(pl.specs(circuit) (0,0) ['resources'])
```

Quantum Teleportation





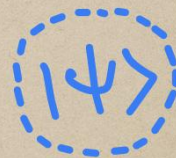
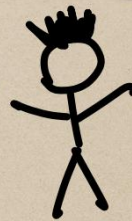
Alice



$|\psi\rangle$



Bob





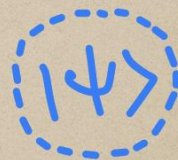
Alice



$|\psi\rangle$



Bob



Open the box? →



Measurement →

many copies of





?



Alice



$|\psi\rangle$

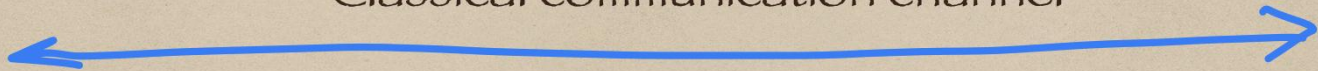


They share an entangled state

Bob



Classical communication channel



Bell states:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$$

$$|\Psi^+\rangle = \frac{1}{\sqrt{2}} (|01\rangle + |10\rangle)$$

$$|\Psi^-\rangle = \frac{1}{\sqrt{2}} (|01\rangle - |10\rangle)$$

$$|\Phi^-\rangle = \frac{1}{\sqrt{2}} (|00\rangle - |11\rangle)$$

They form a basis for 2-qubit space



We can measure in Bell basis

Ex: $P = |\Phi^+\rangle \langle \Phi^+|$

They are maximally entangled



Teleportation protocol



Alice

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|0_A 0_B\rangle + |1_A 1_B\rangle)$$





Teleportation protocol

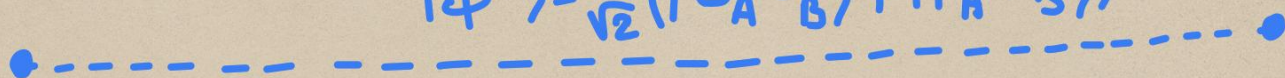


Alice

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$



$$|\phi^+\rangle = \frac{1}{\sqrt{2}}(|0_A 0_B\rangle + |1_A 1_B\rangle)$$



Joint system :

$$|\psi\rangle \otimes |\phi^+\rangle = (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle \otimes |0\rangle + |1\rangle \otimes |1\rangle)$$



Teleportation protocol



Alice

Alice measures her 2 qubits in Bell basis



What happens?

Remember

$$|00\rangle = \frac{1}{\sqrt{2}}(|\Phi^+\rangle + |\Phi^-\rangle) \quad , \dots \quad \text{and so on for other basis}$$

Now let's rewrite

$$\begin{aligned} |\psi\rangle \otimes |\Phi^+\rangle &= \frac{1}{\sqrt{2}} (\alpha |00\rangle |0\rangle + \alpha |01\rangle |1\rangle + \beta |10\rangle |0\rangle + \beta |11\rangle |1\rangle) \\ &= \frac{1}{2} (|\Phi^+\rangle (\alpha |0\rangle + \beta |1\rangle) + |\Psi^+\rangle (\alpha |1\rangle + \beta |0\rangle) \\ &\quad + |\Psi^-\rangle (\alpha |1\rangle - \beta |0\rangle) + |\Phi^-\rangle (\alpha |0\rangle - \beta |1\rangle)) \end{aligned}$$

Remember

$$|00\rangle = \frac{1}{\sqrt{2}}(|\Phi^+\rangle + |\Phi^-\rangle) \quad , \dots \quad \text{and so on for other basis}$$

Now let's rewrite

$$\begin{aligned} |\psi\rangle \otimes |\Phi^+\rangle &= \frac{1}{\sqrt{2}} (\alpha |00\rangle |0\rangle + \alpha |01\rangle |1\rangle + \beta |10\rangle |0\rangle + \beta |11\rangle |1\rangle) \\ &= \frac{1}{2} (|\Phi^+\rangle (\alpha |0\rangle + \beta |1\rangle) + |\Psi^+\rangle (\alpha |1\rangle + \beta |0\rangle) \\ &\quad + |\Psi^-\rangle (\alpha |1\rangle - \beta |0\rangle) + |\Phi^-\rangle (\alpha |0\rangle - \beta |1\rangle)) \end{aligned}$$

Alice measures and gets one of these four states.

Let's look at the possible results

Alia $|\Phi^+\rangle$

00

$|\Psi^+\rangle$
01

$|\Phi^-\rangle$

10

$|\Phi^-\rangle$

11

Bob

$\alpha|10\rangle + \beta|11\rangle$

$\alpha|11\rangle + \beta|10\rangle$

$\alpha|11\rangle - \beta|10\rangle$

$\alpha|10\rangle - \beta|11\rangle$

Let's look at the possible results

Alice $|\Phi^+\rangle$

00

$|\Psi^+\rangle$
01

$|\Psi^-\rangle$
10

$|\Phi^-\rangle$
11

Bob

$\alpha|10\rangle + \beta|11\rangle \quad \checkmark \rightarrow I$

$\alpha|11\rangle + \beta|10\rangle \rightarrow X \rightarrow \checkmark$

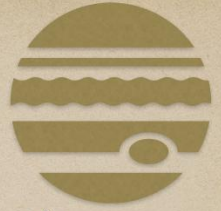
$\alpha|11\rangle - \beta|10\rangle \rightarrow Y \rightarrow \checkmark$

$\alpha|10\rangle - \beta|11\rangle \rightarrow Z \rightarrow \checkmark$

But Bob doesn't know yet!



Teleportation protocol



Alice

Alice measures her 2 qubits in Bell basis

Alice needs to send 2 classical bits that shows the outcome of her measurement

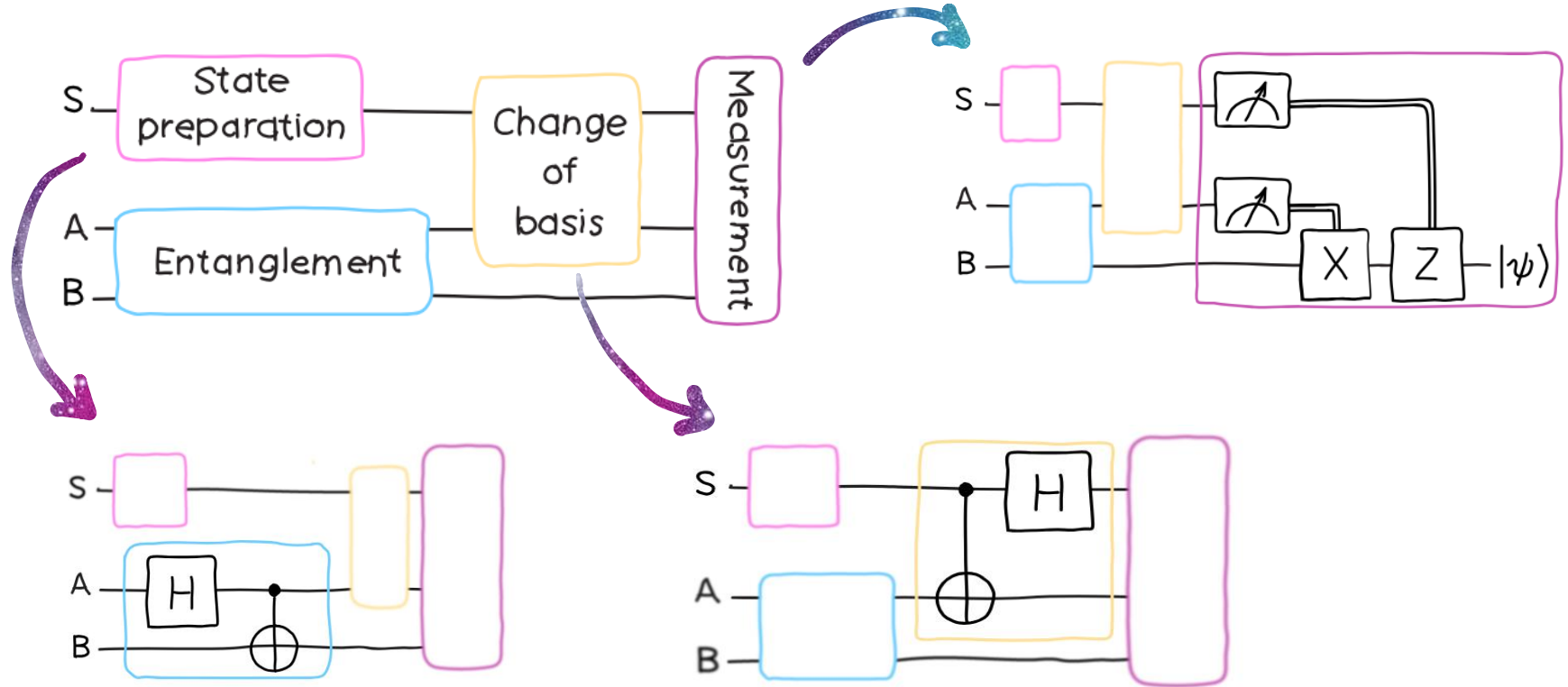


Bob applies the corrections (I,X,Y,Z)
based on Alice's outcome

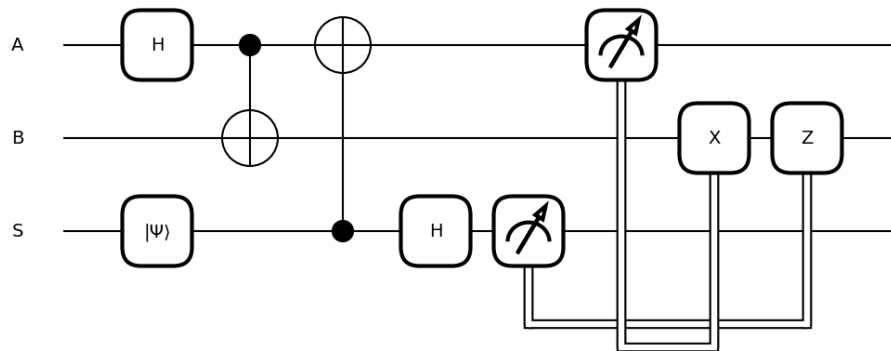
Fun questions you should be asking about teleportation

- Is the state unknown for Bob too? Yes!
- Is the state copied? No. Because of no-cloning theorem!
- Why not just communicate the coefficient over the classical channel? Because it's expensive! We would need exponential resources to do that
- When Alice measure, did the information travel faster than the speed of light? No! we have no-signaling
So something more interesting is happening here

Quantum teleportation as a circuit



Exercise 5: Teleportation



```
def state_preparation(state):
    # use pl.StatePrep to prepare wire S in the given state

def entangle_qubits():
    # use pl.Hadamard and pl.CNOT to create a Bell pair

def basis_rotation():
    # use pl.Hadamard and pl.CNOT to rotate the basis

def measure_and_update():
    # use pl.measure for Alice's measurement and pl.PauliX, pl.PauliZ, and pl.cond for Bob's correction

def teleport(state):
    state_preparation(state)
    entangle_qubits()
    basis_rotation()
    measure_and_update()

state = np.array([1 / np.sqrt(2) + 0.3j, 0.4 - 0.5j])
_ = pl.draw_mpl(teleport)(state)
```